

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-099477

(43)Date of publication of application : 07.04.2000

(51)Int.Cl.

G06F 15/16

G06F 12/14

G06F 13/00

(21)Application number : 10-266141

(71)Applicant : FUJI XEROX CO LTD

(22)Date of filing : 21.09.1998

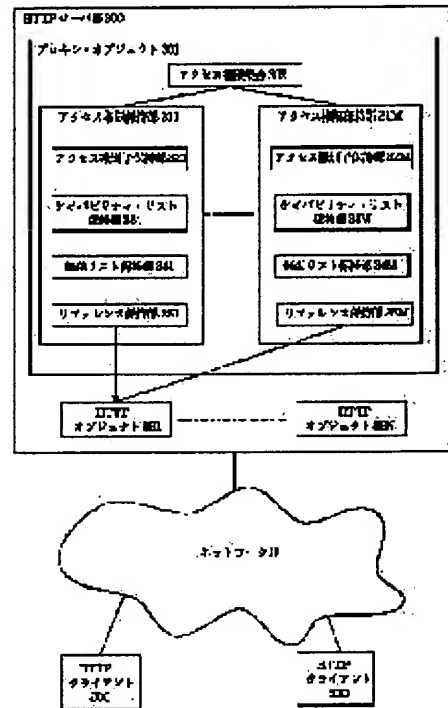
(72)Inventor : HORIKIRI KAZUNORI

## (54) METHOD FOR MANAGING ACCESS TO OBJECT

(57)Abstract:

**PROBLEM TO BE SOLVED:** To safely protect objects existing on a network by providing the method with a step for connecting 1st authority information to an access identifier (ID) by an access management server and enciphering the connected information by applying a public key owned by the server itself to generate a 1st access key and other steps.

**SOLUTION:** A proxy object 301 generates an access ID and authority information for the ID. Then the object 301 connects the authority information to the access ID and enciphers the connected information by a public key to generate an access key. An HTTP client 50C receives the access key by message exchange using a cipher such as a secure socket layer(SSL) to/from an HTTP server 300. The client 50C receiving the access key connects authority information prepared by itself to a random number generated by itself and enciphers the connected information by the public key to generate an access key.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-99477

(P2000-99477A)

(43) 公開日 平成12年4月7日(2000.4.7)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テーマコード(参考)
G 0 6 F 15/16	6 2 0	G 0 6 F 15/16	6 2 0 T
12/14	3 1 0	12/14	3 1 0 K
13/00	3 5 1	13/00	3 5 1 Z

審査請求 未請求 請求項の数16 O L (全 23 頁)

(21) 出願番号 特願平10-266141

(22) 出願日 平成10年9月21日(1998.9.21)

(71) 出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂二丁目17番22号

(72) 発明者 堀切 和典

神奈川県足柄上郡中井町境430 グリーン

テクなかい 富士ゼロックス株式会社内

(74) 代理人 100086531

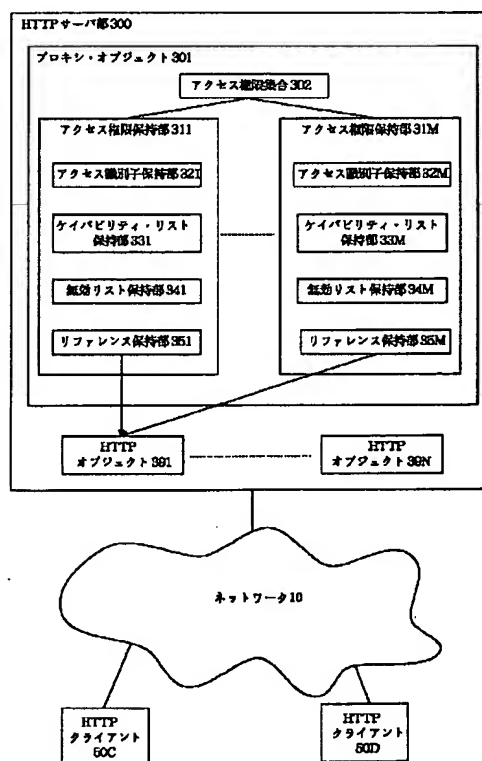
弁理士 澤田 俊夫

(54) 【発明の名称】 オブジェクトのアクセス管理方法

(57) 【要約】

【課題】 権限内容を変更したケイパビリティをクライアントが自由に生成し、他のクライアントに安全に委譲することができる、オブジェクトのアクセス管理方式を提供する。

【解決手段】 アクセス識別子やケイパビリティなどの秘密情報は、暗号鍵若しくは一方向性関数によって暗号化された形態でネットワーク上を伝送されるので、オブジェクトにアクセスするためのアクセス・キーが不正に使用される可能性は極めて低い。また、新たに生成した権限情報をアクセス・キーに連結して暗号化することによって新たなアクセス・キーを派生することを許容しているので、権限情報のバリエーションを増大させることができる。ケイパビリティの保持者には、さらに派生したアクセス・キーを無効化する権限も与えられる。



## 【特許請求の範囲】

【請求項1】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上におけるオブジェクトのアクセス管理方法であって、(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成するステップと、(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子aとを連結し、自身が持つ公開鍵暗号系の公開鍵 $pkey$ を適用して暗号化し、第1のアクセス・キー $akey_1 = pkey(P_1, a)$ を生成するステップと、(c) 第1のアクセス・キー $akey_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第N-1のアクセス・キー $akey_{N-1}$ を持つオブジェクト・クライアントが、第Nの権限情報 $P_N$ を生成して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ とを連結して、公開鍵 $pkey$ を適用して暗号化し、第Nのアクセス・キー $akey_N = pkey(P_N, akey_{N-1})$ を生成するステップと(但し、Nは正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(e) オブジェクト・クライアントが、第Nのアクセス・キー $akey_N$ を提示してオブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、自身が持つ公開鍵暗号系の秘密鍵 $skey$ を用いて第Nのアクセス・キー $akey_N$ を復号化して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ を得て、これらを検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法。

【請求項2】オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 公開鍵暗号系の公開鍵 $pkey$ と秘密鍵 $skey$ を保持する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成する手段と、

(c) 第1の権限情報 $P_1$ とアクセス識別子aとを連結し、公開鍵 $pkey$ を適用して暗号化し、第1のアクセス・キー $akey_1 = pkey(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $akey_1$ をオブジェクト・クライアントに配布する手段と、(e) 第Nのアクセス・キー $akey_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第Nのアクセ

ス・キー $akey_N$ は、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ を連結し、公開鍵 $pkey$ を適用して生成されたアクセス・キー $pkey(P_N, akey_{N-1})$ のことであり、Nは正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、

(f) 秘密鍵 $skey$ を用いて第Nのアクセス・キー $akey_N$ を復号化して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ を得て、これらを検査する手段と、(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバ。

【請求項3】前記手段(f)は、最初のアクセス識別子aが現れるまで、秘密鍵 $skey$ を用いた復号化処理を再帰的に実行することを特徴とする請求項2に記載のアクセス管理サーバ。

【請求項4】前記手段(f)は、秘密鍵 $skey$ を用いた復号化処理によって順次得られた各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査することを特徴とする請求項3に記載のアクセス管理サーバ。

【請求項5】さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第Mのアクセス・キー $akey_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、

前記手段(f)は、秘密鍵 $skey$ を用いて第Nのアクセス・キー $akey_N$ を再帰的に復号化処理する途中で無効化されたアクセス・キー $akey_N$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶することを特徴とする請求項3又は4のいずれかに記載のアクセス管理サーバ。

【請求項6】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、公開鍵暗号系の公開鍵 $pkey$ と秘密鍵 $skey$ を保持するものとする)、(a) アクセス識別子aと第1の権限情報 $P_1$ を与えられたオブジェクトAについての第N-1のアクセス・キー $akey_{N-1}$ を受信する手段と、(b) 第Nの権限情報 $P_N$ を生成する手段と、(c) 第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ とを連結して、公開鍵 $pkey$ を適用して暗号化し、第Nのアクセス・キー $akey_N = pkey(P_N, akey_{N-1})$ を生成する手段と、(d) 第Nのアクセス・キー $akey_N$ を用いてオブジェクトAへのアクセスを要求し、第Nのアクセス・キー $akey_N$ を他のオブジェクト・クライアントに委譲し、又は、第Nのアクセス・キー $akey_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト

ト・クライアント。

【請求項7】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システムにおけるオブジェクトのアクセス管理方法であって、(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子 $a$ と、アクセス識別子 $a$ に対する第1の権限情報 $P_1$ を生成するステップと、(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子 $a$ に対して一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成するステップと、(c) 第1のアクセス・キー $key_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第 $N-1$ のアクセス・キー $key_{N-1}$ を持つオブジェクト・クライアントが、第 $N$ の権限情報 $P_N$ を生成し、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して一方向性関数 $f$ を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成するステップと(但し、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子 $a$ とする。)、(e) オブジェクト・クライアントが、第 $N$ のアクセス・キー $key_N$ と、各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示して、オブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、オブジェクトAのアクセス識別子 $a$ と受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して一方向性関数 $f$ を順次適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法。

【請求項8】オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システムにおいて、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 一方向性関数 $f$ を提供する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子 $a$ と、アクセス識別子 $a$ に対する第1の権限情報 $P_1$ を生成する手段と、(c) 第1の権限情報 $P_1$ とアクセス識別子 $a$ に対して一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $key_1$ をオブジェクト・クライアントに配布する手段と、(e) 第 $N$ のアクセス・キー $key_N$ と各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第 $N$ のアクセス・キー $key_N$

$key_N$ は、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して一方向性関数 $f$ を適用して生成されたアクセス・キー $f(P_N, key_{N-1})$ のことであり、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子 $a$ とする。)、(f) オブジェクトAのアクセス識別子 $a$ と受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して一方向性関数 $f$ を順次適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査する手段と、(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバ。

【請求項9】前記手段(f)は、受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査することを特徴とする請求項8に記載のアクセス管理サーバ。

【請求項10】さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第 $M$ のアクセス・キー $key_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、

前記手段(f)は、一方向性関数 $f$ を順次適用して第 $N$ のアクセス・キー $key_N$ を生成する途中で無効化されたアクセス・キー $key_M$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶する、ことを特徴とする請求項9に記載のアクセス管理サーバ。

【請求項11】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システムにおいて、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、一方向性関数 $f$ を提供するものとする)、(a) アクセス識別子 $a$ と第1の権限情報 $P_1$ を与えられたオブジェクトAについての第 $N-1$ のアクセス・キー $key_{N-1}$ と各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を受信する手段と、(b) 第 $N$ の権限情報 $P_N$ を生成する手段と、(c) 第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して一方向性関数を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成する手段と、(d) 第 $N$ のアクセス・キー $key_N$ に各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を添えて、オブジェクトAへのアクセスを要求し、第 $N$ のアクセス・キー $key_N$ を他のオブジェクト・クライアントに委譲し、又は、第 $N$ のアクセス・キー $key_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト・クライアント。

【請求項12】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス管理サーバとがネットワーク

を介して接続された分散型ネットワーク・システム上におけるオブジェクトのアクセス管理方法であって、

(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成するステップと、

(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子aに対して可換な一方向性関数 $f$ を適用して、第1のアクセス・キー $akey_1 = f(P_1, a)$ を生成するステップと、(c) 第1のアクセス・キー $akey_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第 $N-1$ のアクセス・キー $akey_{N-1}$ を持つオブジェクト・クライアントが、第 $N$ の権限情報 $P_N$ を生成し、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $akey_{N-1}$ に対して可換な一方向性関数 $f$ を適用して、第 $N$ のアクセス・キー $akey_N = f(P_N, akey_{N-1})$ を生成するステップと(但し、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(e) オブジェクト・クライアントが、第 $N$ のアクセス・キー $akey_N$ と、各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示して、オブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、オブジェクトAのアクセス識別子aと受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して可換な一方向性関数 $f$ を任意の順番で適用して、改めて第 $N$ のアクセス・キー $akey_N$ を生成し、これと受信した第 $N$ のアクセス・キー $akey_N$ とを比較して検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法。

【請求項13】オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 可換な一方向性関数 $f$ を提供する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成する手段と、(c) 第1の権限情報 $P_1$ とアクセス識別子aに対して可換な一方向性関数 $f$ を適用して、第1のアクセス・キー $akey_1 = f(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $akey_1$ をオブジェクト・クライアントに配布する手段と、(e) 第 $N$ のアクセス・キー $akey_N$ と各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第 $N$ のアクセス・キー $akey_N$ は、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $akey_{N-1}$ に対して可換な一方向性関数 $f$ を適用して生成されたアクセス・キー $f(P_N, akey_{N-1})$ のことであり、 $N$ は正の整数であり、第

0のアクセス・キーはアクセス識別子aとする。)、

(f) オブジェクトAのアクセス識別子aと受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して可換な一方向性関数 $f$ を任意の順番で適用して、改めて第 $N$ のアクセス・キー $akey_N$ を生成し、これと受信した第 $N$ のアクセス・キー $akey_N$ とを比較して検査する手段と、(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバ。

10 【請求項14】前記手段(f)は、受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査することを特徴とする請求項13に記載のアクセス管理サーバ。

【請求項15】さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第 $M$ のアクセス・キー $akey_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、前記手段(f)は、可換な一方向性関数 $f$ を任意の順番で適用して第 $N$ のアクセス・キー $akey_N$ を生成する途中で無効化されたアクセス・キー $akey_M$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶する、ことを特徴とする請求項13に記載のアクセス管理サーバ。

【請求項16】オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、可換な一方向性関数 $f$ を提供するものとする)、(a) アクセス識別子aと第1の権限情報 $P_1$ を与えられたオブジェクトAについての第 $N-1$ のアクセス・キー $akey_{N-1}$ と各権限情報 $P_1, P_2, \dots, P_{N-1}$ を受信する手段と、(b) 第 $N$ の権限情報 $P_N$ を生成する手段と、(c) 第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $akey_{N-1}$ に対して可換な一方向性関数を適用して、第 $N$ のアクセス・キー $akey_N = f(P_N, akey_{N-1})$ を生成する手段と、(d) 第 $N$ のアクセス・キー $akey_N$ に各権限情報 $P_1, P_2, \dots, P_{N-1}$ を添えて、オブジェクトAへのアクセスを要求し、第 $N$ のアクセス・キー $akey_N$ を他のオブジェクト・クライアントに委譲し、又は、第 $N$ のアクセス・キー $akey_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト・クライアント。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ネットワーク上に複数のコンピュータ・システムが相互接続された分散型のネットワーク・システムにおいて、ネットワーク上に存在するオブジェクトを安全に保護するためのオブジェ

クトのアクセス管理方式に係り、特に、オブジェクトのアクセス権限を記述したケイパビリティ (Capability) をクライアントに配ることによってオブジェクトへのアクセスを許可するタイプのアクセス管理方式に関する。

【0002】また、本発明は、ネットワーク・システム上のクライアント間でオブジェクトへのアクセス権限を与えることができるオブジェクトのアクセス管理方式に係り、特に、アクセス権限を記述したケイパビリティを安全に委譲したりケイパビリティを無効化することができるオブジェクトのアクセス管理方式に関する。

【0003】

【従来の技術】昨今の情報処理・情報通信の分野における発展は目覚ましいものがある。この種の技術分野においては、コンピュータ・システム同士を相互接続する、すなわち「コンピュータ・ネットワーキング」に関する研究・開発は、従来より活発に行われてきた。システム同士を相互接続する主な目的は、複数ユーザによるコンピュータ資源の共有や、情報の共有・流通などである。

【0004】システム間を接続するための伝送媒体すなわち「ネットワーク」としては、大学や事業所の構内など限られた空間内に敷設されたLAN (Local Area Network) の他、LANを専用回線で接続したWAN (Wide Area Network) や、一般公衆回線 (PSTN)、ISDN (Integrated Service Digital Network)、インターネットなど様々である。

【0005】ネットワーク・システムは、一般に、ネットワーク上の特定のコンピュータをサーバ (ファイル・サーバ、プリント・サーバ) とし、これを他のクライアントが利用し合うというクライアント・サーバ・モデルとして構築される。また、ネットワーク上においては、各コンピュータ間でアプリケーションを分散処理する分散環境が実現されている。

【0006】また、ネットワーク上には、無数のオブジェクトが存在する。例えばサーバはサービス・オブジェクトを所有し、これに対し、クライアントは、遠隔手続呼び出し (Remote Procedure Call: RPC)、又は、遠隔メソッド呼び出し (Remote Method Invocation: RMI) などを用いて遠隔的にサービスを要求することができる。ネットワークがTCP/IP (Transmission Control Protocol/Internet Protocol) 接続で構成されるインターネットであり、サーバがHTTP (Hypertext Transfer Protocol) サーバであれば、HTTPサーバが所有するサービス・オブジェクトの実体は、HTTPクライアントにおいてホームページを形成するためのHTML (Hypertext Markup Language) ファイルである。

HTTPクライアントは、例えばURL (Uniform Resource Locator) によってサービス要求を行うことができる。

【0007】ところで、オブジェクトに対するアクセスは、一般には、オブジェクト自身が持つアクセス識別子を参照することによって実現される。アクセス識別子は、例えばオブジェクトの名前 (オブジェクト識別子) であることもある。例えば、オブジェクトの実体がファイルであれば、識別子としてのファイル名をファイル・システムに提示することによりファイル・アクセスを果たすことができる。また、HTTPサーバ・クライアントで構成されるWWW空間では、URLがオブジェクトの識別子に相当する。アクセス識別子は、1つのオブジェクトに対して一対一だけではなく、1つのオブジェクトが複数のアクセス識別子を用意することもある。

【0008】他方、最近では、オブジェクトへのアクセスを制限という技術に関して研究開発が進められている。特に、ネットワーク上にオブジェクトが散在し、複数のクライアントがアクセスを試みるような分散ネットワーク環境においては、セキュリティの観点からもアクセス制御は非常に重要な課題である。

【0009】オブジェクトに対するアクセスを制限するための方式の1つとして、「ケイパビリティ (Capability)」方式がある。ケイパビリティとは、オブジェクトに対するアクセス権限のことである。例えば、アクセスが許可されている有効期限や、許可された残りアクセス回数、許可された操作 (読み出し、書き込み、実行など) などがケイパビリティに含まれる項目である。ケイパビリティ方式によれば、オブジェクトは、自身に用意された複数のアクセス識別子の各々に対して異なったケイパビリティを与えることもできる。例えば、あるアクセス識別子a1に対しては有効アクセス回数を10回に設定する一方、他のアクセス識別子a2には3回のアクセスに制限したりする。あるいは、アクセス識別子a1には読み出し、書き込み、実行、削除など全ての操作権限を与えるが、アクセス識別子a2には読み出ししか許さないようにすることもできる。

【0010】ケイパビリティは、オブジェクトへのアクセスを許容する一種の鍵の役割を持つ。すなわち、ケイパビリティを所有するクライアントは、オブジェクトへのアクセスを許可される。例えば、ネットワークが無数のコンピュータ・システムがTCP/IP接続されてなるインターネットのような場合には、ケイパビリティをURL文字列の中で記述することもできる。

【0011】ところで、クライアント・ユーザは、ネットワーク上に散在する有益なリソースすなわちオブジェクトを、他のユーザと共有することを望むことがある。例えば、クライアントA (上司) が、自分の不在中に、クライアントB (部下) にオブジェクトの管理を委ねたいような場合である。この場合、クライアントAは、他

のクライアントBに対してケイパビリティを委譲することによって、ある1つのオブジェクトを共有することができる。

【0012】ところが、クライアントが、自分と同じケイパビリティを他人に与えてしまえば、オブジェクトが思わぬ不利益を被ることがある。上記の例で言えば、クライアントAが、クライアントBに自分と同じケイパビリティを委譲してしまうと、クライアントAが職場に復帰した後もクライアントBはオブジェクトにアクセスし続けることができる。また、クライアントBはオブジェクトの読み出しオペレーションだけを許可すれば充分なのに、クライアントAと等しく書き込みや実行オペレーションまで不用意に許可してしまえば、オブジェクトに対して不測の改竄が加えられかねない。

【0013】要言すれば、ネットワーク・システムの運用上、ケイパビリティの委譲を認めるべきではある一方、ケイパビリティの委譲を無制限に許すとネットワークやオブジェクトのセキュリティは脅かされかねないのである。委譲の際には、操作権限に所定の制限を加えるなど、ケイパビリティを弱める必要がある。

【0014】また、ケイパビリティを不正に改竄されないためにも、ケイパビリティの譲受者に対しては、ケイパビリティの内容すら秘匿すべきである。ネットワーク上では、オブジェクトに対するアクセス識別子やそのケイパビリティは、そもそも、秘密情報としての性格が強い。

【0015】更に言えば、オブジェクトへのアクセスを管理するサーバは、クライアントが生成したケイパビリティを安全且つ確実に検査する必要がある。

【0016】また、ケイパビリティを生成したクライアントは、オブジェクトが危険にさらされるなどの所定の事態の検知若しくは予見によって、生成したケイパビリティを無効化する必要もあるであろう。

【0017】例えば特開平5-81204号公報には、分散型コンピュータ・システムにおけるアクセス制御について開示されている。同公報によれば、ケイパビリティに相当する特権属性証明書(PAC)の代理使用を制御すると同時に、PACを多くの目的のために使用できる方法が提供されている。すなわち、PACを配布する際には、これに開始者実体に相当する開始者資格付与属性を含めるようにするとともに、暗号的に開始者資格付与属性を有した暗号キーを開始者実体に配布するようになっている。

【0018】しかしながら、特開平5-81204号公報に開示された方法によれば、使用期限やアクセス回数など、ケイパビリティのバリエーションには対応できない。また、開始実体者がケイパビリティを弱めたPACを自由に作成する方法については一切言及していない。

【0019】また、特開平9-319659号公報に

は、非分散型のコンピュータ・システムにおいて、各ユーザに異なるケイパビリティを割り当てる方式について開示されている。同公報によれば、コンピュータ機能の全体はケイパビリティ・セットを有するイベント・セットに細分化されており、ユーザがコンピュータ・システム上で実行しようとする特定のジョブに応じてケイパビリティが与えられるようになっている。しかしながら、特開平9-319659号公報に係る発明は、ケイパビリティを安全に守ることができない分散環境に適用されたものではない。また、同公報は、オブジェクト間でケイパビリティの検査を安全に行う方法については一切言及していない。

【0020】また、特開平9-251425号公報には、分散システムにおけるシステム資源へのアクセスのセキュリティ制御について開示している。すなわち、グループ識別標識を記憶しておき、これをターゲット・オブジェクトに結び付け、次にメンバーシップ検査を使用して、ターゲット・オブジェクトへのアクセスを要求するクライアントがターゲットに対するアクセス権限を有するグループ・メンバーであるか否かを判断するようになっている。

【0021】しかしながら、特開平9-251425号公報では、使用期限やアクセス回数など、ケイパビリティのバリエーションには対応する手法については開示していない。また、あるケイパビリティの保有者が新たにケイパビリティを生成したりこれを無効化する手法についても一切言及していない。

【0022】また、分散OS(オペレーティング・システム)として当業界では周知の"Amoeba"は、譲渡可能で且つ偽造が困難なケイパビリティを提供している。これは以下の機構により実現される。

(1) クライアントがオブジェクト生成要求をサーバに送信する。

(2) サーバは、オブジェクトを生成して、オブジェクト番号と乱数を割り当てる。乱数は、オブジェクト番号でインデックス付けされたオブジェクト・テーブルに格納される。

(3) サーバは、乱数をキーとして権限と乱数を暗号化したチェック・フィールドを含むケイパビリティを生成する。

(4) ケイパビリティをクライアントに返す。

(5) クライアントが、サーバに対して、ケイパビリティを提示したアクセス要求を出す。

(6) サーバは、ケイパビリティからオブジェクト番号とチェック・フィールドを抽出し、オブジェクト・テーブルに格納されている乱数を基にチェック・フィールドから権限と乱数を復号化する。

(7) 復号の結果得られた乱数がオブジェクト・テーブルに格納された乱数と一致するかどうか検証する。

(8) 要求された操作がケイパビリティに記述された権



限に合致すれば、サーバはその操作を実行する。

【0023】Amoebaでは、さらに、ケイパビリティの保持者が権限を弱めた新しいケイパビリティを生成するために、以下の機構も備えている。

(1) クライアントとサーバがN個の可換な一方向性関数を共有する。

(2) クライアントが第2の権限を生成する。

(3) クライアントが第1の権限と第2の権限との差分の権限に該当する番号と対応付けた一方向性関数を全て適用して、第2のチェック・フィールドを生成する。

(4) サーバが、第2の権限と第2のチェック・フィールドを含むケイパビリティを受信する。

(5) サーバは、権限フィールドに従い、一方向性関数を順次適用して、クライアントが提示したチェック・フィールドと一致した場合に、その操作を実行する。

【0024】しかしながら、Amoebaは、N個の可換な一方向性関数を用いてN個の権限の可否を記述する方法については規定しているが、使用権限や回数といった権限についての多くのバリエーションに対応する点については考慮していない。また、ケイパビリティの無効化に関しては、サーバで保持されている乱数を変更することによってあるオブジェクトについての全てのケイパビリティを無効化する手法が規定されているが、特定のケイパビリティを無効化する手法については言及していない。例えば、あるケイパビリティの保持者が生成したケイパビリティや、これから派生したケイパビリティのみを無効化する点は一切規定していない。

【0025】また、Amoebaではオブジェクトにアクセスするためのオブジェクト番号自体については一切プロテクトがかけられていない。

【0026】また、Web上で公開されたBjorn N. Freeman-Bensonの論文"Using the Web to Private Information—or A Short Paper About Password Protection Without Client Modification" (URLは"http://www1.cern.ch/www94/PrelimProcs.html")では、機密性のあるURL(すなわちケイパビリティ)の取り扱いについて開示されている。同論文に記載された方式は以下の手順に従う。

(1) サーバがログイン名とパスワードの組からなるパスワード・リストを生成・保持する。また、サーバは暗号鍵を保持している。

(2) クライアントがログイン名とパスワードの組からなるメッセージをサーバに送信する。

(3) サーバは、パスワード・リストを検索し、クライアントから送信されたログイン名とパスワードの組が見つければ、このログイン名とパスワードを暗号鍵で暗号化した文字列をアクセス・キーとして、URLに含めて

クライアントに送信する。

(4) クライアントは、受け取ったURLを用いてサーバにアクセス要求する。

(5) サーバは、受信したURLからアクセス・キーを抽出し、暗号鍵を用いて復号化して、もとのログイン名とパスワードの組を得る。そして、これがパスワード・リストに登録されているかどうかを検査する。

(6) 登録されていれば、サーバは対応するオブジェクトへのアクセスを許可する。

【0027】また、同論文では、クライアントがパスワードの変更を行うことでURLを無効化することができる点を言及している。

【0028】しかしながら、同論文では、ある1つのオブジェクトについての正当で且つ異なるURL(すなわちケイパビリティ)を複数生成する点については開示していない。また、パスワードの保持者ではなくURL(すなわちケイパビリティ)の保持者がURLを無効化する点についても一切言及していない。

【0029】

【発明が解決しようとする課題】本発明の目的は、ネットワーク上に複数のコンピュータ・システムが相互接続された分散型のネットワーク・システムにおいて、ネットワーク上に存在するオブジェクトを安全に保護することができる、優れたオブジェクトのアクセス管理方式を提供することにある。

【0030】本発明の更なる目的は、オブジェクトのアクセス権限を記述したケイパビリティ(Capability)をクライアントに配ることによってオブジェクトへのアクセスを許可するタイプの、優れたアクセス管理方式を提供することにある。

【0031】本発明の更なる目的は、ケイパビリティを保有するクライアントが権限を変更したケイパビリティを自由に生成し、他のクライアントに安全に委譲することができる、優れたオブジェクトのアクセス管理方式を提供することにある。

【0032】本発明の更なる目的は、ケイパビリティを保有するクライアントが権限を変更したケイパビリティを自由に生成し、オブジェクトを管理するサーバは生成されたケイパビリティを安全に検査することができる、優れたオブジェクトのアクセス管理方式を提供することにある。

【0033】本発明の更なる目的は、ケイパビリティを保有するクライアントが権限を変更したケイパビリティを自由に生成し、さらに生成したケイパビリティを確実に無効化することができる、オブジェクトのアクセス管理方式を提供することにある。

【0034】

【課題を解決するための手段】本発明は、上記課題を参酌してなされたものであり、その第1の側面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オ



ブジェクトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上におけるオブジェクトのアクセス管理方法であって、(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成するステップと、(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子aとを連結し、自身が持つ公開鍵暗号系の公開鍵 $pk_{key}$ を適用して暗号化し、第1のアクセス・キー $akey_1 = pk_{key}$

( $P_1, a$ )を生成するステップと、(c) 第1のアクセス・キー $akey_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第N-1のアクセス・キー $akey_{N-1}$ を持つオブジェクト・クライアントが、第Nの権限情報 $P_N$ を生成して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ とを連結して、公開鍵 $pk_{key}$ を適用して暗号化し、第Nのアクセス・キー $akey_N = pk_{key}(P_N, akey_{N-1})$ を生成するステップと(但し、Nは正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(e) オブジェクト・クライアントが、第Nのアクセス・キー $akey_N$ を提示してオブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、自身が持つ公開鍵暗号系の秘密鍵 $sk_{key}$ を用いて第Nのアクセス・キー $akey_N$ を復号化して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ を得て、これらを検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法である。

【0035】また、本発明の第2の側面は、オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 公開鍵暗号系の公開鍵 $pk_{key}$ と秘密鍵 $sk_{key}$ を保持する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成する手段と、(c) 第1の権限情報 $P_1$ とアクセス識別子aとを連結し、公開鍵 $pk_{key}$ を適用して暗号化し、第1のアクセス・キー $akey_1 = pk_{key}(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $akey_1$ をオブジェクト・クライアントに配布する手段と、(e) 第Nのアクセス・キー $akey_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第Nのアクセス・キー $akey_N$ は、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$

$akey_{N-1}$ を連結し、公開鍵 $pk_{key}$ を適用して生成されたアクセス・キー $pk_{key}(P_N, akey_{N-1})$ のことであり、Nは正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(f) 秘密鍵 $sk_{key}$ を用いて第Nのアクセス・キー $akey_N$ を復号化して、第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ を得て、これらを検査する手段と、(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバである。

【0036】ここで、前記手段(f)は、最初のアクセス識別子aが現れるまで、秘密鍵 $sk_{key}$ を用いた復号化処理を再帰的に実行するようにしてもよい。

【0037】また、前記手段(f)は、秘密鍵 $sk_{key}$ を用いた復号化処理によって順次得られた各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査するようにしてもよい。

【0038】また、アクセス管理サーバは、さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第Mのアクセス・キー $akey_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、前記手段(f)は、秘密鍵 $sk_{key}$ を用いて第Nのアクセス・キー $akey_N$ を再帰的に復号化処理する途中で無効化されたアクセス・キー $akey_M$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶するようにしてもよい。

【0039】また、本発明の第3の側面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、公開鍵暗号系の公開鍵 $pk_{key}$ と秘密鍵 $sk_{key}$ を保持するものとする)、(a) アクセス識別子aと第1の権限情報 $P_1$ を与えられたオブジェクトAについての第N-1のアクセス・キー $akey_{N-1}$ を受信する手段と、(b) 第Nの権限情報 $P_N$ を生成する手段と、

(c) 第Nの権限情報 $P_N$ と第N-1のアクセス・キー $akey_{N-1}$ とを連結して、公開鍵 $pk_{key}$ を適用して暗号化し、第Nのアクセス・キー $akey_N = pk_{key}(P_N, akey_{N-1})$ を生成する手段と、(d) 第Nのアクセス・キー $akey_N$ を用いてオブジェクトAへのアクセスを要求し、第Nのアクセス・キー $akey_N$ を他のオブジェクト・クライアントに委譲し、又は、第Nのアクセス・キー $akey_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト・クライアントである。

【0040】また、本発明の第4の面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェ

10

20

30

40

50

クトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上におけるオブジェクトのアクセス管理方法であって、(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成するステップと、(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子aに対して一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成するステップと、(c) 第1のアクセス・キー $key_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第 $N-1$ のアクセス・キー $key_{N-1}$ を持つオブジェクト・クライアントが、第 $N$ の権限情報 $P_N$ を生成し、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して一方向性関数 $f$ を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成するステップと(但し、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(e) オブジェクト・クライアントが、第 $N$ のアクセス・キー $key_N$ と、各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示して、オブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、オブジェクトAのアクセス識別子aと受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して一方向性関数 $f$ を順次適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法である。

【0041】また、本発明の第5の側面は、オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 一方向性関数 $f$ を提供する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子aと、アクセス識別子aに対する第1の権限情報 $P_1$ を生成する手段と、(c) 第1の権限情報 $P_1$ とアクセス識別子aに対して一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $key_1$ をオブジェクト・クライアントに配布する手段と、(e) 第 $N$ のアクセス・キー $key_N$ と各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第 $N$ のアクセス・キー $key_N$ は、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対

して一方向性関数 $f$ を適用して生成されたアクセス・キー $key_N = f(P_N, key_{N-1})$ のことであり、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子aとする。)、(f) オブジェクトAのアクセス識別子aと受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して一方向性関数 $f$ を順次適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査する手段と、

(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバである。

【0042】ここで、前記手段(f)は、受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査するようにしてもよい。

【0043】また、アクセス管理サーバは、さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第 $M$ のアクセス・キー $key_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、前記手段(f)は、一方向性関数 $f$ を順次適用して第 $N$ のアクセス・キー $key_N$ を生成する途中で無効化されたアクセス・キー $key_M$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶するようにしてもよい。

【0044】また、本発明の第6の側面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、一方向性関数 $f$ を提供するものとする)、(a) アクセス識別子aと第1の権限情報 $P_1$ を与えられたオブジェクトAについての第 $N-1$ のアクセス・キー $key_{N-1}$ と各権限情報 $P_1, P_2, \dots, P_{N-1}$ を受信する手段と、(b) 第 $N$ の権限情報 $P_N$ を生成する手段と、

(c) 第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して一方向性関数を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成する手段と、(d) 第 $N$ のアクセス・キー $key_N$ に各権限情報 $P_1, P_2, \dots, P_{N-1}$ を添えて、オブジェクトAへのアクセスを要求し、第 $N$ のアクセス・キー $key_N$ を他のオブジェクト・クライアントに委譲し、又は、第 $N$ のアクセス・キー $key_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト・クライアントである。

【0045】また、本発明の第7の側面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトを要求する1以上のオブジェクト・クライアントと、あるオブジェクトへのアクセスを管理するアクセス

管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上におけるオブジェクトのアクセス管理方法であって、(a) アクセス管理サーバが、オブジェクトAにアクセスするためのアクセス識別子 $a$ と、アクセス識別子 $a$ に対する第1の権限情報 $P_1$ を生成するステップと、(b) アクセス管理サーバが、第1の権限情報 $P_1$ とアクセス識別子 $a$ に対して可換な一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成するステップと、(c) 第1のアクセス・キー $key_1$ がオブジェクト・クライアントに配布されるステップと、(d) 第 $N-1$ のアクセス・キー $key_{N-1}$ を持つオブジェクト・クライアントが、第 $N$ の権限情報 $P_N$ を生成し、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して可換な一方向性関数 $f$ を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成するステップと(但し、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子 $a$ とする。)、(e) オブジェクト・クライアントが、第 $N$ のアクセス・キー $key_N$ と、各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示して、オブジェクトAへのアクセスを要求するステップと、(f) アクセス管理サーバが、オブジェクトAのアクセス識別子 $a$ と受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して可換な一方向性関数 $f$ を任意の順番で適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査するステップと、(g) アクセス管理サーバが、前記(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容するステップと、を含むことを特徴とするオブジェクトのアクセス管理方法である。

【0046】また、本発明の第8の側面は、オブジェクトを提供する1以上のオブジェクト・サーバとオブジェクトを要求する1以上のオブジェクト・クライアントとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを管理するためのアクセス管理サーバであって、(a) 可換な一方向性関数 $f$ を提供する手段と、(b) オブジェクトAにアクセスするためのアクセス識別子 $a$ と、アクセス識別子 $a$ に対する第1の権限情報 $P_1$ を生成する手段と、(c) 第1の権限情報 $P_1$ とアクセス識別子 $a$ に対して可換な一方向性関数 $f$ を適用して、第1のアクセス・キー $key_1 = f(P_1, a)$ を生成する手段と、(d) 第1のアクセス・キー $key_1$ をオブジェクト・クライアントに配布する手段と、(e) 第 $N$ のアクセス・キー $key_N$ と各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ を提示したオブジェクトAへのアクセス要求を受信する手段と(但し、第 $N$ のアクセス・キー $key_N$ は、第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して可換な一方向性関数 $f$ を適用して生

成されたアクセス・キー $f(P_N, key_{N-1})$ のことであり、 $N$ は正の整数であり、第0のアクセス・キーはアクセス識別子 $a$ とする。)、(f) オブジェクトAのアクセス識別子 $a$ と受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ に対して可換な一方向性関数 $f$ を任意の順番で適用して、改めて第 $N$ のアクセス・キー $key_N$ を生成し、これと受信した第 $N$ のアクセス・キー $key_N$ とを比較して検査する手段と、(g) 前記手段(f)における検査が成功したことに応答して、オブジェクトAへのアクセスを許容する手段と、を含むことを特徴とするアクセス管理サーバである。

【0047】ここで、前記手段(f)は、受信した各権限情報 $P_1, P_2, \dots, P_{N-1}, P_N$ のコンテキストをも検査するようによい。

【0048】また、アクセス管理サーバは、さらに無効化されたアクセス・キーを保持する無効アクセス・キー・テーブルと、オブジェクト・クライアントから無効化要求された第 $M$ のアクセス・キー $key_M$ を前記無効アクセス・キー・テーブルに登録する手段とを含み、前記手段(f)は、可換な一方向性関数 $f$ を任意の順番で適用して第 $N$ のアクセス・キー $key_N$ を生成する途中で無効化されたアクセス・キー $key_M$ を得ると、検査を失敗させ、オブジェクトAへのアクセスを拒絶するようにしてもよい。

【0049】また、本発明の第9の側面は、オブジェクトを提供する1以上のオブジェクト・サーバと、オブジェクトへのアクセスを管理するアクセス管理サーバとがネットワークを介して接続された分散型ネットワーク・システム上において、オブジェクトへのアクセスを要求するクライアントであって(但し、アクセス管理サーバは、可換な一方向性関数 $f$ を提供するものとする)、

(a) アクセス識別子 $a$ と第1の権限情報 $P_1$ を与えられたオブジェクトAについての第 $N-1$ のアクセス・キー $key_{N-1}$ と各権限情報 $P_1, P_2, \dots, P_{N-1}$ を受信する手段と、(b) 第 $N$ の権限情報 $P_N$ を生成する手段と、(c) 第 $N$ の権限情報 $P_N$ と第 $N-1$ のアクセス・キー $key_{N-1}$ に対して可換な一方向性関数 $f$ を適用して、第 $N$ のアクセス・キー $key_N = f(P_N, key_{N-1})$ を生成する手段と、(d) 第 $N$ のアクセス・キー $key_N$ に各権限情報 $P_1, P_2, \dots, P_{N-1}$ を添えて、オブジェクトAへのアクセスを要求し、第 $N$ のアクセス・キー $key_N$ を他のオブジェクト・クライアントに委譲し、又は、第 $N$ のアクセス・キー $key_N$ の無効化を要求する手段と、を含むことを特徴とするオブジェクト・クライアントである。

【0050】

【作用】本発明に係るオブジェクトのアクセス管理方式によれば、アクセス識別子やキーパリティなどの秘密情報は、暗号鍵若しくは一方向性関数によって暗号化された形態でネットワーク上を伝送される。したがって、

オブジェクトにアクセスするためのアクセス・キーが解読され、不正に使用される可能性は極めて低い。

【0051】また、新たに生成した権限情報をアクセス・キーに連結して暗号化することによって新たなアクセス・キーを派生することを許容しているので、権限情報のバリエーションを増大させることができる。

【0052】また、権限情報を生成したケイパビリティの保持者が、さらにケイパビリティを弱めた権限情報を連結して新たなアクセス・キーを生成することができるので、アクセス権限の委譲を安全に行うことができる。ケイパビリティの保持者には、さらに派生したアクセス・キーを無効化する権限も与えられる。

【0053】本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

#### 【0054】

##### 【発明の実施の形態】1. 第1の実施形態

本発明は、オブジェクトに対するアクセスを制御するためのものである。特に、複数のコンピュータ・システムがネットワークを介して接続されたネットワーク・システム上において、ネットワーク上に存在するオブジェクトに対するアクセスを制御するためのものである。

【0055】ここで言うネットワークは、主として、各コンピュータ・システムでアプリケーションを分散処理する分散型ネットワークである。また、ネットワークに接続された各コンピュータ・システムは、オブジェクトを所有するオブジェクト・サーバ（以下、単に「サーバ」とする）と、オブジェクトへのアクセスを要求するオブジェクト・クライアント（以下、単に「クライアント」とする）とに2分される。

【0056】ネットワーク上には、さらにオブジェクトへのアクセス管理（すなわちオブジェクトのセキュリティ管理）を行うためのアクセス管理サーバも存在する。アクセス管理サーバは、オブジェクト・サーバ自身がその機能を兼ね備えることができるし、あるいはネットワーク上の別のコンピュータ・システムにアクセス管理を委ねることもできる。

【0057】アクセス管理サーバは、オブジェクトへのアクセスを管理するために、オブジェクトに対して1以上のアクセス識別子を生成する。アクセス識別子を提示することによってオブジェクトへのアクセスが許可される。

【0058】アクセス識別子は、通常、所定の規則性に従って生成される。例えば識別子中に通し番号が含まれていれば、他のアクセス識別子を容易に推測することができ、オブジェクトのセキュリティが危うくなる。このため、アクセス管理サーバは、アクセス識別子そのものをクライアントに配布することはせずに、元のアクセス識別子に何らかの加工（例えば暗号化）を施したアクセス・キーを代わりに配布するようにしている。アクセス

識別子ではなくアクセス・キーを提示することによって、オブジェクトへのアクセスが許可される。

【0059】また、セキュリティの観点からオブジェクトへのアクセス権限を制限する必要もある。アクセス権限とは、例えば許可されたアクセス回数（残りアクセス回数）や、アクセスの有効期限、許可されたオブジェクトへの操作（読み出し、書きこみ、削除、実行など）などである。当業界では、アクセスに関する権限情報のことを「ケイパビリティ（Capability）」とも呼ぶ。

【0060】アクセス管理サーバは、アクセス識別子を生成するときには、そのアクセス識別子に与えられたケイパビリティも定義する。また、1つのオブジェクトに対して複数のアクセス識別子を生成することもできる。アクセス管理サーバは、各アクセス識別子とそのケイパビリティを、対応するオブジェクトと関連付けて保管するようになっている。また、アクセス識別子から派生したアクセス・キーを保管してもよい。また、アクセス管理サーバは、公開鍵暗号系の公開鍵 *public key* と秘密鍵 *secret key* を有している。

【0061】1. 1 アクセス管理サーバでの操作  
アクセス管理サーバは、まず、あるオブジェクトAへのアクセスを許可するためのアクセス識別子 *a* を生成する。

【0062】また、アクセス管理サーバは、アクセス識別子 *a* に対して第1の権限情報 *P<sub>1</sub>* を生成する。アクセス管理サーバは、アクセス識別子 *a* を第1の権限情報 *P<sub>1</sub>* と関連付けて保管する。

【0063】次いで、アクセス管理サーバは、アクセス識別子 *a* と第1の権限情報 *P<sub>1</sub>* を連結（例えばビット連結）し、公開鍵 *public key* を適用することで暗号化して、第1のアクセス・キー *key<sub>1</sub> (P<sub>1</sub>, a)* を生成する。また、アクセス管理サーバは、アクセス・キー *key<sub>1</sub> (P<sub>1</sub>, a)* を、オブジェクトAと対応付けて保管しておく。第1の権限情報 *P<sub>1</sub>* は、公開鍵 *public key* で暗号化されているので、アクセス管理サーバ以外は解読することができない。

【0064】1. 2 クライアントでの操作  
生成された第1のアクセス・*key<sub>1</sub> = key (P<sub>1</sub>, a)* は、例えばネットワーク経由で、あるクライアントに委譲されたとする。第1のアクセス・キーはアクセス管理サーバの公開鍵 *public key* で暗号化されているので、これを使用するクライアントは原則として、自身に与えられた権限情報 *P<sub>1</sub>* の内容を判読できない。

【0065】アクセス・キーを受け取ったクライアントは、アクセス・キーに対して後に変更を加えることに備えて、サーバの公開鍵 *public key* も取得しておく。

【0066】1. 3 新たなアクセス・キーの生成  
クライアントは、さらに別のクライアントにアクセス・キーを委譲したいことがある。例えば、あるクライアン

トAは、自分の不在中に別のクライアントBにオブジェクトの管理を委ねたいときにはアクセス・キーを渡しておけばよい。

【0067】ところが、自分と同じ権限情報を持つアクセス・キーをクライアントBに渡してしまったならば、思わぬ不利益を被ることがある。例えば、同じ有効期限のままでアクセス・キーを渡してしまつては、復帰後もクライアントBがオブジェクトにアクセスし続けることができる。また、クライアントBはオブジェクトの読み出しオペレーションだけを許可すれば充分なのに、クライアントAと等しく書き込みや実行オペレーションまで許可してしまつては、オブジェクトのセキュリティは脅かされかねない。すなわち、クライアントAは、自分と同じ権限を与えてもよいとは限らず、権限を弱めなければならない場合もある。例えばアクセス・キーの有効期限を制限するとか、操作権限を読み出しに限定して書き込みや実行を禁止する等である。

【0068】このような場合、クライアントは、他のクライアントに与えてもよい権限情報を自ら作成する。これを第2の権限情報 $P_2$ としておく。そして、クライアントは、第1のアクセス・キー $pkey(P_1, *$

$$akey_N = pkey(P_N, akey_{N-1})$$

【0072】1. 4 オブジェクトへのアクセス  
第Nのアクセス・キー $akey_N$ を持つクライアントは、オブジェクトAを管理するアクセス管理サーバに対して、このアクセス・キーを用いてオブジェクトAへのアクセスを要求することができる。アクセス要求は、アクセス・キーを含んだ要求メッセージの形式で送信される。

【0073】これに対し、アクセス管理サーバは、アクセス・キー $akey_N$ が正当なものがどうかを検証しなければならない。

【0074】アクセス管理サーバは、要求メッセージを受信すると、その中に含まれるアクセス・キー $akey_N$ を取り出す。そして、サーバ自身が持つ公開鍵暗号系の秘密鍵 $key$ を用いてこのアクセス・キーを復元する。上式が $akey_N = pkey(P_N, akey_{N-1})$ 成立することから、この復元作業によって、第Nの権限情報 $P_N$ と第N-1番目のアクセス・キー $akey_{N-1}$ が得られる。

【0075】アクセス管理サーバは、得られたアクセス・キー $akey_{N-1}$ が保管しているアクセス識別子 $a$ と一致すれば、復元処理を終了する。一致しなければ、さらに秘密鍵 $key$ を用いた復元処理を試みる。このような復元処理は、アクセス・キー $akey_{N-1}$ とアクセス識別子 $a$ との照合が成功裏に終わるまで再帰的に試行される。

【0076】再帰的な復元処理の結果、アクセス管理サーバは、一連の権限情報 $P_N, P_{N-1}, \dots, P_2, P_1$ も得

\*  $a$ )と第2の権限情報 $P_2$ を連結(例えばビット連結)し、さらに公開鍵 $pkey$ を適用することで第2のアクセス・キー $pkey(P_2, pkey(P_1, a))$ を生成する。

【0069】第2のアクセス・キーは、公開鍵 $pkey$ で暗号化されているので、第2の権限情報 $P_2$ を作成したクライアント自身と、秘密鍵 $key$ を持つサーバ以外は、第2のアクセス・キーに含まれた権限情報 $P_2$ の内容を解読できない。例えば第2のアクセス・キーを譲り受けたクライアントも、基本的には、自分に与えられた権限情報 $P_2$ を確認できない。

【0070】以下では、アクセス識別子 $a$ から第N番目に派生したアクセス・キーのことを「第Nのアクセス・キー」 $akey_N$ と呼び、また、第N番目のアクセス・キー $akey_N$ に与えられた権限情報を第Nの権限情報 $P_N$ と呼ぶことにする(但し、Nは0以上の整数であり、第0のアクセス・キーはアクセス識別子 $a$ とする)。第Nのアクセス・キーと第N-1のアクセス・キーの間には、下式の関係が成立する。

【0071】

【数1】

$$akey_N = pkey(P_N, akey_{N-1})$$

る。アクセス管理サーバは、この一連の権限情報のコンテキストを検証してもよい。アクセス管理サーバは、例えば、アクセス・キーを作成する各クライアントに対して、権限を弱める方向でしか権限情報の作成を認めないように規定することができる。権限を弱めるとは、例えばアクセス回数を減らすとか、有効期限を短縮するとか、操作を制限することを意味する。あるいは、アクセス回数のみ権限の強化を認める、などのようにコンテキストを規定してもよい。

【0077】一連の権限情報のコンテキストを検証した結果、コンテキストが所定の規則に反していれば、アクセス管理サーバは、要求メッセージ中のアクセス・キー $akey_N$ を不正とみなすこともできる。

【0078】アクセス管理サーバは、アクセス・キーとオブジェクトAのアクセス識別子との照合と、権限情報のコンテキスト検証に成功したときのみ、要求メッセージ中のアクセス・キー $akey_N$ が正当であると判断する。そして、該当するオブジェクトAへのアクセスを認める。

【0079】1. 5 アクセス・キーの無効化

アクセス管理サーバは、一度正当なものとして認められたアクセス・キーを無効化する機構を備えてもよい。この機構は、無効化されたアクセス・キーを登録するための無効アクセス・キー・テーブルを備えることで実現される。

【0080】クライアントは、無効化したいアクセス・キーを含めた無効化要求メッセージを送信する。アクセ

ス管理サーバは、無効化要求メッセージを受信すると、アクセス・キーを取り出して、これを無効アクセス・キー・テーブルに追加登録すればよい。

【0081】アクセス管理サーバは、オブジェクトAへのアクセス要求メッセージを受信すると、アクセス・キーを取り出し、秘密鍵 *key* を用いてアクセス・キーを再帰的に復元していく（上述）。この再帰的な復元工程の途中で、無効アクセス・キー・テーブルに登録されているアクセス・キーと同じアクセス・キーが出現すると、要求メッセージが不正であると判断して、該当するオブジェクトAへのアクセスを拒否する。

#### 【0082】2. 第2の実施形態

第1の実施形態では、アクセス・キーを生成するために、アクセス管理サーバが持つ公開鍵暗号系の公開鍵と暗号鍵を用いた。第2の実施形態では、暗号鍵の代わりに2個の引数を持つ方向性関数を用いることにする。ここで言う「方向性関数」とは、逆関数を求めることが極めて困難な関数のことであり、該関数を適用する前の引数の値を推測することを不可能にする作用がある。アクセス管理サーバとクライアントは2引数 (*x*, *y*) の方向性関数  $f(x, y)$  を共有するものとする。

【0083】2. 1 アクセス管理サーバでの操作  
アクセス管理サーバは、まず、あるオブジェクトAへのアクセスを許可するためのアクセス識別子 *a* を生成する。

【0084】また、アクセス管理サーバは、アクセス識別子 *a* に対して第1の権限情報  $P_1$  を生成する。アクセス管理サーバは、アクセス識別子を権限情報と対応付けて保管する。第1の権限情報  $P_1$  とは、アクセス識別子 *a* を提示したものに与えられる権限すなわちプライバシーのことであり（前述）。

【0085】次いで、アクセス管理サーバは、アクセス識別子 *a* と第1の権限情報  $P_1$  に方向性関数  $f$  を適用して、第1のアクセス・キー  $f(P_1, a)$  を生成する。アクセス管理サーバは、第1のアクセス・キー  $f(P_1, a)$  を、オブジェクトAと対応付けて保管しておく。

【0086】2. 2 クライアントでの操作  
生成された第1のアクセス・キー  $f(P_1, a)$  は、例えばネットワーク経由で、オブジェクトAへのアクセスが許されたクライアントに送信される。正当なアクセス・キー  $f(P_1, a)$  を持つクライアントは、これを提示することでオブジェクトAへのアクセスが許される。

【0087】アクセス・キーを受け取ったクライアントは、アクセス・キーの権限情報に対して後で変更を加えることに備えて、アクセス管理サーバが使用する方向性関数も取得しておく。

【0088】2. 3 新たなアクセス・キーの生成  
クライアントは、さらに別のクライアントにアクセス・キーを委譲したいことがある（上述）。

【0089】このような場合、クライアントは、他のクライアントに与えてもよいプライバシーを自ら作成する。これを第2の権限情報  $P_2$  としておく。そして、クライアントは、第1のアクセス・キー  $f(P_1, a)$  と第2の権限情報  $P_2$  に対してさらに方向性関数  $f$  を適用して、第2のアクセス・キー  $f(P_2, f(P_1, a))$  を生成する。

【0090】方向性関数  $f$  のパラメータ値は、アクセス管理サーバも解読できない。このため、第2のアクセス・キーに第2の権限情報  $P_2$  を添付して第2のアクセス・キーを委譲するようにする。

【0091】以下では、アクセス識別子 *a* から第N番目に派生したアクセス・キーのことを「第Nのアクセス・キー」 $key_N$  と呼び、また、第N番目のアクセス・キー  $key_N$  に与えられた権限情報を第Nの権限情報  $P_N$  と呼ぶことにする（但し、Nは0以上の整数であり、第0のアクセス・キーはアクセス識別子 *a* とする）。第Nのアクセス・キーと第N-1のアクセス・キーの間には、下式の関係が成立する。

【0092】

【数2】

$$key_N = f(P_N, key_{N-1})$$

【0093】方向性関数  $f$  を一度適用すると、権限情報  $P_N$  は解読不能となってしまふ。このため、アクセス・キーを作成したクライアントは、権限情報  $P_N$  を添えてアクセス・キー  $f(P_N, key_{N-1})$  を配布するようにする。この結果、第Nのアクセス・キーを送信するメッセージ中には、アクセス・キーに含まれる全ての権限情報  $P_1, P_2, \dots, P_{N-1}, P_N$  が同封されることになる。

【0094】2. 4 オブジェクトへのアクセス  
第Nのアクセス・キー  $key_N$  を持つクライアントは、オブジェクトAを管理するアクセス管理サーバに対して、このアクセス・キーを用いてオブジェクトAへのアクセスを要求することができる。

【0095】但し、アクセス管理サーバは方向性関数  $f$  を適用された第Nのアクセス・キー  $key_N$  を解読できない。このため、クライアントは、アクセス・キー  $key_N$  を受信したときに添付されていた全ての権限情報  $P_1, P_2, \dots, P_{N-1}, P_N$  を沿えて、要求メッセージの形式で送信する。

【0096】これに対し、アクセス管理サーバは、アクセス・キー  $key_N$  が正当なものがどうかを検証を行う。

【0097】アクセス管理サーバは、要求メッセージを受信すると、その中に含まれるアクセス・キー  $key_N$  と全ての権限情報  $P_1, P_2, \dots, P_{N-1}, P_N$  を取り出す。そして、方向性関数  $f$  を用いて、第1のアクセス・キー  $key_1 (= f(P_1, a))$ 、第2のアクセス



・キー  $key_2 (= f(P_2, f(P_1, a)))$ ,  
 …, 第Nのアクセス・キー  $key_N (= f(P_N, key_{N-1}))$  を順次生成していく。

【0098】そして、アクセス管理サーバは、自ら生成した第Nのアクセス・キー  $key_N$  が、要求メッセージに含まれていたアクセス・キー  $key_N$  と一致すれば、アクセス・キーの照合は成功したと判断する。

【0099】また、アクセス管理サーバは、要求メッセージに含まれている一連の権限情報についてのコンテキストの検証(上述)を行ってもよい。

【0100】アクセス管理サーバは、アクセス・キー  $key_N$  の検査と、権限情報のコンテキスト検証に成功したときのみ、要求メッセージ中のアクセス・キー  $key_N$  が正当であると判断する。そして、対応するオブジェクトへのアクセスを認める。

【0101】2. 5 アクセス・キーの無効化

アクセス管理サーバは、一度正当なものとして認められたアクセス・キーを無効化する機構を備えてもよい。この機構は、無効化されたアクセス・キーを登録するための無効アクセス・キー・テーブルを備えることで実現される。

【0102】クライアントは、無効化したいアクセス・キーを含めた無効化要求メッセージを送信する。アクセス管理サーバは、無効化要求メッセージを受信すると、\*

$$f(x, f(y, a)) = f(y, f(x, a))$$

【0107】アクセス管理サーバとクライアントは可換な一方向性関数  $f(x, y)$  を共有するものとする。

【0108】3. 1 アクセス管理サーバでの操作  
 アクセス管理サーバは、まず、あるオブジェクトAへのアクセスを許可するためのアクセス識別子  $a$  を生成する。

【0109】また、アクセス管理サーバは、アクセス識別子  $a$  に対して第1の権限情報  $P_1$  を生成する。アクセス管理サーバは、アクセス識別子を権限情報と対応付けて保管する。第1の権限情報  $P_1$  とは、アクセス識別子  $a$  を提示したものに与えられる権限すなわちケイパビリティのことである(前述)。

【0110】次いで、アクセス管理サーバは、アクセス識別子  $a$  と第1の権限情報  $P_1$  に可換な一方向性関数  $f$  を適用して、第1のアクセス・キー  $f(P_1, a)$  を生成する。アクセス管理サーバは、第1のアクセス・キー  $f(P_1, a)$  を、オブジェクトAと対応付けて保管しておく。

【0111】3. 2 クライアントでの操作  
 生成された第1のアクセス・キー  $f(P_1, a)$  は、例えばネットワーク経由で、オブジェクトAへのアクセスが許されたクライアントに送信される。正当なアクセス・キー  $f(P_1, a)$  を持つクライアントは、これを提示することでオブジェクトAへのアクセスが許される。

\* アクセス・キーを取り出して、これを無効アクセス・キー・テーブルに追加登録すればよい。

【0103】アクセス管理サーバは、オブジェクトAへのアクセス要求メッセージを受信すると、アクセス・キーと  $key_N$  と全ての権限情報  $P_1, P_2, \dots, P_{N-1}, P_N$  を取り出す。そして、一方向性関数  $f$  を用いて、第1のアクセス・キー  $key_1 (= f(P_1, a))$ , 第2のアクセス・キー  $key_2 (= f(P_2, f(P_1, a)))$ , …, 第Nのアクセス・キー  $key_N (= f(P_N, key_{N-1}))$  を順次生成していく(上述)。

【0104】このアクセス・キーの生成工程の途中で、無効アクセス・キー・テーブルに登録されているアクセス・キーが出現すると、要求メッセージが不正であると判断して、該当するオブジェクトAへのアクセスを拒否する。

【0105】3. 第3の実施形態

第2の実施形態では、アクセス・キーを生成するために、2個の引数を持つ一方向性関数を用いた。第3の実施形態では、この一方向性関数が可換であるとする。可換な一方向性関数とは、下式が成立する一方向性関数のことである。

【0106】

【数3】

【0112】アクセス・キーを受け取ったクライアントは、アクセス・キーの権限情報に対して後で変更を加えることに備えて、アクセス管理サーバが使用する可換な一方向性関数  $f$  も取得しておく。

【0113】3. 3 新たなアクセス・キーの生成  
 クライアントは、さらに別のクライアントにアクセス・キーを委譲したいことがある(上述)。

【0114】このような場合、クライアントは、他のクライアントに与えてもよいケイパビリティを自ら作成する。これを第2の権限情報  $P_2$  としておく。そして、クライアントは、第1のアクセス・キー  $f(P_1, a)$  と第2の権限情報  $P_2$  に対してさらに可換な一方向性関数  $f$  を適用して、第2のアクセス・キー  $f(P_2, f(P_1, a))$  を生成する。

【0115】可換な一方向性関数  $f$  のパラメータ値は、アクセス管理サーバも解読できない。このため、第2のアクセス・キーに第2の権限情報  $P_2$  を添付して第2のアクセス・キーを委譲するようにする。

【0116】以下では、アクセス識別子  $a$  から第N番目に派生したアクセス・キーのことを「第Nのアクセス・キー」  $key_N$  と呼び、また、第N番目のアクセス・キー  $key_N$  に与えられた権限情報を第Nの権限情報  $P_N$  と呼ぶことにする(但し、Nは0以上の整数であり、第0のアクセス・キーはアクセス識別子  $a$  とす



る)。第Nのアクセス・キーと第N-1のアクセス・キーの間には、下式の関係が成立する。

【0117】

【数4】

$$a k e y_N = f(P_N, a k e y_{N-1})$$

【0118】可換な一方向性関数fを一度適用すると、権限情報P<sub>N</sub>は解読不能となってしまう。このため、アクセス・キーを作成したクライアントは、権限情報P<sub>N</sub>を添えてアクセス・キーf(P<sub>N</sub>, a k e y<sub>N-1</sub>)を配布するようにする。この結果、第Nのアクセス・キーを送信するメッセージ中には、アクセス・キーに含まれる全ての権限情報P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>N-1</sub>, P<sub>N</sub>が同封されることになる。

【0119】3. 4 オブジェクトへのアクセス  
第Nのアクセス・キーa k e y<sub>N</sub>を持つクライアントは、オブジェクトAを管理するアクセス管理サーバに対して、このアクセス・キーを用いてオブジェクトAへのアクセスを要求することができる。

【0120】但し、アクセス管理サーバは可換な一方向性関数fを適用された第Nのアクセス・キーa k e y<sub>N</sub>を解読できない。このため、クライアントは、アクセス・キーa k e y<sub>N</sub>を受信したときに添付されていた全ての権限情報P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>N-1</sub>, P<sub>N</sub>を沿えて、要求メッセージの形式で送信する。

【0121】これに対し、アクセス管理サーバは、アクセス・キーa k e y<sub>N</sub>が正当なものがどうかを検証を行う。

【0122】アクセス管理サーバは、要求メッセージを受信すると、その中に含まれるアクセス・キーa k e y<sub>N</sub>と全ての権限情報P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>N-1</sub>, P<sub>N</sub>を取り出す。そして、可換な一方向性関数fを用いて、各アクセス・キーを自ら生成する。但し、可換な一方向性関数fは順序性を守る必要がないので、各アクセス・キーを任意の順番で生成することができる。

【0123】そして、アクセス管理サーバは、自ら生成した第Nのアクセス・キーa k e y<sub>N</sub>が、要求メッセージに含まれていたアクセス・キーa k e y<sub>N</sub>と一致すれば、アクセス・キーの照合は成功したと判断する。

【0124】また、アクセス管理サーバは、要求メッセージに含まれている一連の権限情報についてのコンテキストの検証(上述)を行ってもよい。

【0125】アクセス管理サーバは、アクセス・キーa k e y<sub>N</sub>の検査と、権限情報のコンテキスト検証に成功したときのみ、要求メッセージ中のアクセス・キーa k e y<sub>N</sub>が正当であると判断する。そして、該当するオブジェクトAへのアクセスを認める。

【0126】なお、第3の実施形態でも、第1及び第2の実施形態と同様にアクセス・キーを無効化するための機構を備えていてもよい。但し、第2の実施形態と略同

一の仕組みにより実現できるので、ここでは敢えて説明しない。

【0127】

【実施例】以下、図面を参照しながら本発明の実施例を詳解する。

【0128】図1には、本発明の実施に供されるネットワーク・システム100の構成を模式的に示している。同図に示すように、ネットワーク・システム100は、データの伝送媒体であるネットワーク10上に、複数のコンピュータ・システム50A, 50B, ...が接続されて構成される。

【0129】ネットワーク10は、例えば大学や企業の構内などの限られた空間内に敷設されたLAN(Local Area Network)である。あるいは、LAN同士を専用線等で相互接続してなるWAN(Wide Area Network)や、一般公衆回線(PSTN:Public Switched Telephone Network)、ISDN(Integrated Service Digital Network)、これらネットワークの大規模な集合体であるインターネットであってもよい。ネットワーク10は、主として、接続された各コンピュータ・システム50A, 50B, ...においてアプリケーションを分散処理する分散型ネットワークである。

【0130】各コンピュータ・システム50A, 50B, ...は、LANアダプタやモデム、TA(Terminal Adapter)等の回線終端装置(DCE:Data Circuit Terminating Equipment)を介してネットワーク10に接続されている。各コンピュータ・システム50A, 50B, ...は、オブジェクトを所有するオブジェクト・サーバと、オブジェクトへのアクセスを要求するオブジェクト・クライアントに2分される。サーバ又はクライアントの機能に特化してデザインされた専用マシンであってもよいが、一般には、サーバ用又はクライアント用のアプリケーションを導入して動作する汎用機でよい。各コンピュータ・システム50A, 50B, ...同士は、ネットワーク10を介して、例えばTCP/IP(Transmission Control Protocol/Internet Protocol)接続されている。

【0131】オブジェクト・サーバ50A, 50Bは、複数のオブジェクトを所有する。オブジェクトの一例は、ホームページを形成するためのHTML(Hypertext Markup Language)ファイルである。HTMLファイルは、HTTP(Hypertext Transfer Protocol)プロトコルに従ってネットワーク10上を伝送することができることから、以下では「HTTPオブジェクト」と呼ぶことにする。一方のオブジェクト・クライアント50

C, 50D, 50Eは、URL (Uniform Resource Locator) の形式でHTTPオブジェクトの場所を指定することができる。

【0132】また、ネットワーク上には、さらにHTTPオブジェクトへのアクセス管理（すなわちオブジェクトのセキュリティ）管理を行うためのアクセス管理サーバも存在する。ネットワーク10上の別のコンピュータ・システムがアクセス管理サーバとして働くことも可能であるが、オブジェクト・サーバ自身がアクセス管理機能を持つことも可能である。後述の各実施例では、オブジェクト・サーバ内の「プロキシ・オブジェクト」がHTTPオブジェクトへのアクセス管理を行うものとする。

【0133】図2には、HTTPオブジェクト・サーバ50Aの構成を模式的に示している。図示しない他のオブジェクト・サーバ50Bも同様の構成を備えていると把握されたい。同図に示すように、オブジェクト・サーバ50Aは、N個のHTTPオブジェクト321～32Nの他、各HTTPオブジェクトへのアクセスを管理するプロキシ・オブジェクト301を含んでいる。オブジェクト・サーバ50AのURLは、SSL (Secure Socket Layer) を用いたHTTPプロトコルに従って“https://www300/”と表されるものとする。（以下の実施例では、ケイパビリティのような秘密情報をネットワーク経由で安全に移送するために、SSLを用いているが、一般には、暗号化を行う他の通信方式、又は、暗号化を行わない通信方式を用いても、本発明の効果を奏することはできる。）

【0134】プロキシ・オブジェクト301は、各HTTPオブジェクトに対するリファレンスすなわちアクセス識別子を生成する。1つのHTTPオブジェクトに対して2以上のアクセス識別子を生成してもよい。同一のHTTPオブジェクトに対する各アクセス識別子には、それぞれ個別にアクセス権限すなわちケイパビリティ (Capability) が与えられる。プロキシ・オブジェクト301は、個々のアクセス権限を管理するために、アクセス識別子毎にアクセス権限保持部を生成する。

【0135】図2に示した例では、あるHTTPオブジェクト391に対して、少なくとも2つのアクセス識別子が生成されており、プロキシ・オブジェクト301は各アクセス識別子に対してそれぞれアクセス権限保持部311及び31Mを生成している。

【0136】アクセス権限保持部311は、アクセス識別子保持部321と、ケイパビリティ・リスト保持部331と、無効リスト保持部341と、リファレンス保持部351とを含んでいる。他のアクセス権限保持部31Mも、アクセス権限保持部311と略同一の構成を有すると把握されたい。

【0137】アクセス識別子保持部311は、与えられ

たアクセス識別子を記憶するユニットである。リファレンス保持部351は、アクセス識別子に対応するHTTPオブジェクト391へのポインタを記憶するユニットである。

【0138】アクセス識別子には、該当するHTTPオブジェクトに対する権限が規定されている。ここで言う権限とは、HTTPオブジェクトに対する操作権限（例えば読み出しのみ、読み書き可、実行可）や、アクセス有効期限、有効アクセス回数などである。これらの権限情報は「ケイパビリティ」と呼ばれる。ケイパビリティは、原初的にはプロキシ・オブジェクト301が与えるが、後述するようにアクセス・キーを付与されたHTTPクライアントが逐次作成する。ケイパビリティ・リスト保持部331は、1つのアクセス識別子から派生した各ケイパビリティの現在の権限内容を保管するユニットである。

【0139】また、アクセス・キーを生成した（又は受け取った）クライアントは、アクセス・キーを無効化することが許容されている。無効リスト保持部341は、1つのアクセス識別子から派生したアクセス・キーのうち無効化されたものを保管するユニットである。

【0140】プロキシ・オブジェクト301内で生成されたアクセス権限保持部311, …, 31Mは、1つのアクセス権限集合302を構成する。

【0141】本実施例では、URLを用いてケイパビリティを表現するとともに、暗号を用いてURLを推測困難な形式に変換することによって、HTTPオブジェクトに対するアクセス保護を実現する。以下、各実施例について説明する。

#### 【0142】1. 第1の実施例

第1の実施例では、各アクセス識別子に関する権限情報のセキュリティを守るために、プロキシ・オブジェクト301が保有する公開鍵暗号系の公開鍵 *key* と秘密鍵 *key* を用いる。

【0143】プロキシ・オブジェクト301は、アクセス識別子 *a* とこれに対する権限情報 *P* を生成する。さらに、プロキシ・オブジェクト301は、権限情報 *P* とアクセス識別子を連結して、公開鍵 *key* で暗号化することによって、アクセス・キー *key<sub>i</sub> = key (P<sub>i</sub>, a)* を生成する。

【0144】HTTPクライアント50Cは、HTTPサーバ300との間で、SSL (Secure Socket Layer) などの暗号を用いたメッセージ交換により、アクセス・キー *key<sub>i</sub>* を受信する。

【0145】アクセス・キー *key<sub>i</sub>* を受信したHTTPクライアント50Cは、自身が作成した権限情報“GET, 2, Apr:24:10:00:48:1998:GMT, Apr:24:10:05:48:1998:GMT”と、自身が発生した乱数 *Rnd1* を連結し、さらにこれを公開鍵 *key* で暗号化することで、

下式〔数5〕に示すアクセス・キー $akey_z$ を生成する。但し、権限内容には、実行することができるメソッドの他に、使用可能な記憶容量の上限やプロセッサ占有時間の上限を含んでいてもよい。また、権限情報を記述\*

**{akey<sub>1</sub>, GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1}pkey**

【0147】アクセス・キー $akey_z$ は、 $akey_1$ と結び付けられたオブジェクトに対して、世界時で1998年の4月24日10時0分48秒から10時5分48秒の間にGETメソッドを2回実行することを許容する旨の権限情報を包含している。

**http://www300/{akey<sub>1</sub>, GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1}pkey**

【0150】そして、HTTPクライアント50Cは、他のHTTPクライアント50DにこのURLを送信する。

【0151】HTTPクライアント50Dは、このURLに基づき、ネットワーク10上のwww300で指定されるホスト（この例ではプロキシ・オブジェクト30★

**GET{akey<sub>1</sub>, GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1}pkey HTTP/1.1**

【0153】HTTPサーバ300は、このメッセージを受信すると、メッセージからメソッドGETと、アクセス・キー $akey_z$ 2に相当する文字列" $akey_1$ , GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1"を抽出して、プロキシ・オブジェクト301に入力する。

【0154】プロキシ・オブジェクト301は、自身の秘密鍵 $skey$ を適用して復号化した結果から、アクセス・キー $akey_z$ に与えられたケイパビリティが" $akey_1$ と結び付けられたオブジェクトに対して、世界時で1998年の4月24日10時0分48秒から10☆

**メソッド: GET**

**残り回数: 1**

**有効期限: Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT**

**アクセス・キー: akey<sub>1</sub>, GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1, pkey**

【0157】ここで、さらにHTTPクライアント50DがHTTPサーバ300に対して〔数7〕で示したものと同一のメッセージを送信したときの動作について考察してみる。

【0158】HTTPサーバ300は、メッセージを受信すると、メッセージからメソッドGETと、アクセス・キー $akey_z$ に相当する文字列" $akey_1$ , GET, 2, Apr: 24: 10: 00: 48: 1998: GMT, Apr: 24: 10: 05: 48: 1998: GMT, Rnd 1"を抽出して、プロキシ・オブジ

\*するために、他の任意の表現形式を用いてもよい。

【0146】

〔数5〕

※【0148】次いで、HTTPクライアント50Cは、アクセス・キー $akey_z$ を用いて下式〔数6〕に示すURLを生成する。

【0149】

〔数6〕

★1が存在するサーバと同一で動作しているHTTPサーバ300と、SSLを用いて接続し、下式〔数7〕で示したメッセージを送信する。

【0152】

〔数7〕

20☆時5分48秒の間にGETメソッドを2回実行することの権利であることを判読する。

【0155】また、プロキシ・オブジェクト301は、世界時での現在時刻を取得する。例えば現在時刻が1998年4月24日10時1分48秒であれば、ケイパビリティ・リスト保持部331に下式〔数8〕に示す内容のケイパビリティを保存するとともに、アクセス・キーに結び付けられたHTTPオブジェクト391に対してメッセージGETを送信する。

【0156】

〔数8〕

ェクト301に入力する。

【0159】プロキシ・オブジェクト301は、アクセス権限集合302に登録されているアクセス・キーを検索して、アクセス・キー $akey_z$ と同一のキーが既に登録されているかどうかを判断する。そして、このアクセス・キー $akey_z$ に該当するケイパビリティの現在の内容が以下の〔数9〕の通りであることを、ケイパビリティ・リスト保持部331から導出する。

【0160】

〔数9〕

メソッド: GET

残り回数: 1

有効期限: Apr.24:10:00:48:1998:GMT, Apr.24:10:05:48:1998:GMT

アクセス・キー: akey<sub>1</sub>, GET, 2, Apr.24:10:00:48:1998:GMT, Apr.24:10:05:48:1998:GMT, Rnd 1, pkey

【0161】プロキシ・オブジェクト301は、世界時での現在時刻を取得する。例えば現在時刻が1998年4月24日10時3分48秒であれば、ケイパビリティ・リスト保持部331中の当該エントリの内容を以下の通りに変更した後、HTTPオブジェクト391に対し\*10

メソッド: GET

残り回数: 0

有効期限: Apr.24:10:00:48:1998:GMT, Apr.24:10:05:48:1998:GMT

アクセス・キー: akey<sub>1</sub>, GET, 2, Apr.24:10:00:48:1998:GMT, Apr.24:10:05:48:1998:GMT, Rnd 1, pkey

【0163】この後、プロキシ・オブジェクト301は、世界時での現在時刻が10時5分48秒を経過した時点で、ケイパビリティ・リスト保持部331中の当該エントリを削除する(削除しないまでも、エントリに対応するケイパビリティは失効する)。

【0164】2. 第2の実施例

第2の実施例では、各アクセス識別子に関する権限情報のセキュリティを守るために、プロキシ・オブジェクト301が保有する一方向性関数fを用いる。「一方向性関数」とは、逆関数を求めることが極めて困難な関数のことであり、該関数を適用する前の引数の値を推測することを不可能にする作用がある。この実施例で用いる一方向性関数は可換であってもよい。

【0165】プロキシ・オブジェクト301は、HTTPオブジェクト391の1つのリファレンスに対して、※30

$cap1 = (f(Rnd0, right1, Rnd1), ((right1, Rnd1)))$

【0169】上式の2項組の第1項は権限情報の不正な改竄を防止するためのチェック・フィールドとして機能し、また、第2項は権限の変更履歴をリスト状に記述している。

【0170】上式に含まれる権限情報right1は、世界時で1998年4月24日0時0分0秒から4月2★

(GET, 5, Apr.24:00:00:00:1998:GMT, Apr.25:00:00:00:1998:GMT)

【0172】但し、権限内容right1には、実行することができるメソッドの他に、使用可能な記憶容量の上限やプロセッサ占有時間の上限を含んでいてもよい。また、権限情報を記述するために、他の任意の表現形式を用いてもよい。

【0173】次いで、プロキシ・オブジェクト301は、サーバ名とアクセス識別子とケイパビリティを含んだ以下の【数13】に示すURL文字列を生成して、HTTPクライアント50Cに送信する。

【0174】

【数13】

<https://www300/a/cap1/>

\*てメッセージGETを送信する。ここで行われるケイパビリティの変更内容は、残り回数を1だけ減分することである。

【0162】

【数10】

※アクセス識別子aを生成し、これをアクセス識別子321で保存する。

【0166】プロキシ・オブジェクト301は、次いで、ケイパビリティ識別子Rnd0を生成し、ケイパビリティ・リスト保持部331でこれを保存する。

【0167】さらに、ケイパビリティ・オブジェクト301は、HTTPクライアントにHTTPオブジェクトへのアクセス(操作)を許可する権限内容を記述した権限情報right1とケイパビリティ識別子Rnd0に対して一方向性関数fを適用して、下式【数11】に示すようなケイパビリティcap1を生成する。cap1は、2項の組のビット列として表される。

【0168】

【数11】

★5日0時0分0秒の間に、GETメソッドを5回実行することを許容する旨を示すものであり、以下の【数12】で示した文字列で構成されているものとする。

【0171】

【数12】

【0175】HTTPクライアント50Cは、HTTPサーバ300との暗号を用いたメッセージ交換により、ケイパビリティcap1を含むURLを受信する。

【0176】以下では、HTTPクライアント50Cが、HTTPサーバ300から取得した権限を弱めて他のHTTP50Dに委譲する操作について説明する。

【0177】HTTPクライアント50Cは、以下の【数14】に示す新しいright2と、ケイパビリティ識別子Rnd2を生成する。

【0178】

【数14】

(GET, 2, Apr.24:10:00:48:1998:GMT, Apr.24:10:05:48:1998:GMT)

【0179】上記の権限情報 `right2` は、世界時で1998年4月24日10時0分48秒から10時5分48秒までの間に、GETメッセージを2回実行することが出来る権限を記述している。HTTPクライアント50Cは、一方向性関数 `f` を用いて、2項組を表すビット

`cap2 = (f(cap1, right1, Rnd2), ((right1, Rnd1), (right2, Rnd2)))`

【0181】次いで、HTTPクライアント50Cは、`cap2`を基に、以下のよ【数16】に示すようなURL文字列を生成して、これを他のHTTPクライアント50Dに送信する。

【0182】

【数16】

`https://www300/a/cap2`

【0183】HTTPクライアント50Dは、HTTPクライアント50Cから受信したURLに従って、ネットワーク上においてwww300で表されるホストで動作中のHTTPサーバ300とSSLを用いて接続して、以下の【数17】に示すメッセージを送信する。

【0184】

【数17】

`GET /a/cap2 HTTP/1.1`

【0185】HTTPサーバ300は、メッセージを受※

《処理手順1》

1: ケイパビリティの第1要素を `cap1` とする。

2: ケイパビリティの第2要素を `list` とする。

3: `cap = Rnd0` とする。

4: `i = 1` とする。

5: `list` の第 `i` 番目の要素を取り出し、その第1要素を `right` とし、第2要素を `Rnd` とする。

6: `cap` が無効リスト保持部に保持されていれば、不正として本処理手順を終了する。そうでなければ、次ステップに進む。

7: `f(cap, right, Rnd)` を計算し、`cap` の新しい値とする。

8: 第 `i` 番目の要素が最後の要素の場合はステップ15に進む。そうでなければ、次ステップに進む

9: `list` の第 `i+1` 番目の要素を取り出し、その第1要素を `rightnext` とし、第2要素を `Rndnext` とする。

10: `right` と `rightnext` を比較し、権限を弱める方向に正しく変更されているかどうかを調べる。正しければ次ステップに進み、正しくなければ不正として本処理手順を終了する。

11: `right = rightnext` とする。

12: `Rnd = Rndnext` とする。

13: `i` を1だけ増分する。

14: ステップ6に戻る。

15: `cap1` と `cap` を比較し、両者が等しければ正しい権限情報とし、等しくなければ不正な権限情報であるとして結果を返す。

\* 列として下式【数15】のようなケイパビリティ `cap2` を生成する。

【0180】

【数15】

※ 信すると、このメッセージの中から、GETメソッドと、アクセス識別子 `a` と、ケイパビリティ `cap2` を抽出する(ケイパビリティ `cap2` は、【数15】に示した文字列で構成される)。そして、HTTPサーバ300は、これらのパラメータをプロキシ・オブジェクト301に入力する。

【0186】プロキシ・オブジェクト301は、アクセス権限集合302の中で入力されたアクセス識別子 `a` を検索して、該当するアクセス権限保持部311内のケイパビリティ・リスト保持部331が保持しているケイパビリティ識別子 `Rnd0` を取得する。

【0187】次いで、プロキシ・オブジェクト301は、以下のような《処理手順1》を実行する。

【0188】

【数18】

【0189】但し、一方向性関数 `f` として、下式のよう ★ 【0190】

な可換性を持つ一方向性関数を用いることもできる。 ★ 【数19】

`f(f(x1, y1, z1), y2, z2) = f(f(x1, y2, z2), y1, z1)`

【0191】可換性のある一方向性関数 `f` を用いること の処理結果が得られる。

により、変更履歴の順番を無視した下記の《処理手順

2》を実行することによっても、《処理手順1》と同様 50 【数20】

【0192】

【数20】

## 《処理手順2》

- 1: ケイパビリティの第1要素をcapfとする。
- 2: ケイパビリティの第2要素をlistとする。
- 3: cap=Rnd0とする。
- 4: listの任意の要素を取り出し、その第1をrightとし、第2要素をRndとし、取り出した要素をlistから取り除く。
- 5: result=rightとする。
- 6: capが無効リスト保持部に保持されていれば不正として本処理手順を終了する。そうでなければ、次ステップに進む。
- 7: f(cap, right, Rnd)を計算し、capの新しい値とする。
- 8: listが空リストの場合はステップ14に進む。そうでなければ次ステップに進む。
- 9: listの任意の要素を取り出し、その第1要素をrightnextとし、第2要素をRndnextとし、取り出した要素をlistから取り除く。
- 10: rightで表される権限とrightnextで表される権限の最低値をとることによって得られる権限を、resultの新しい値とする。
- 11: right = rightnextとする。
- 12: Rnd = Rndnextとする。
- 13: ステップ8に戻る。
- 14: capfとcapを比較し、両者が等しければ正しい権限情報とし、等しくなければ不正な権限情報として結果を返す。

【0193】可換性のある一方向性関数fを用いること 20  
は、ケイパビリティの第2項である変更履歴を表すリストの保持方法として、履歴の順を保持する必要がないことを意味する。例えば、履歴の順番に依らず、権限で許される操作毎にリストを分割して保持したり、権限を特定のビット・フィールドに対応させてリストを保持することが可能となる。

【0194】プロキシ・オブジェクト301は、《処理手順2》を実行した結果から、メッセージに包含されたケイパビリティの内容は「cap1と結び付けられたオブジェクトに対して、世界時刻で1998年の4月24 30  
日10時0分48秒から10時5分48秒の間に、GETメソッドを2かい実行することのできる権利」であると判断する。

【0195】さらに、プロキシ・オブジェクト301は、ケイパビリティ識別子Rnd1とRnd2をキーとして、ケイパビリティ・リスト保持部331内を検索する。該当するエントリがリスト中にないので、世界時の現在時刻を取得し、1998年4月24日10時1分48秒、すなわち与えられたケイパビリティの有効期限内であることから、ケイパビリティ・リスト保持部33 40  
1中に以下のエントリを保存する。

【0196】

【数21】

ケイパビリティ識別子: Rnd1  
メソッド: GET  
残り回数: 4  
ケイパビリティ識別子: Rnd2  
メソッド: GET  
残り回数: 1

【0197】そして、プロキシ・オブジェクト301は、該当するHTTPオブジェクト391に対してメッセージGETを送信するとともに、オブジェクト391から受信した応答を要求元のHTTPクライアントに送信する。

【0198】ここで、HTTPクライアント50DがHTTPサーバ300に対して【数17】に示したのと同じメッセージを送信した場合の動作について説明する。

【0199】プロキシ・オブジェクト301は、上述の《処理手順1》または《処理手順2》に従ってメッセージ中の権限情報を検証した後、受信したケイパビリティcap2に含まれているケイパビリティ識別子Rnd1とRnd2をキーとしてケイパビリティ・リスト保持部331を検索し、該当するエントリから以下に示す履歴を抽出する。

【0200】

【数22】

ケイパビリティ識別子: Rnd1

メソッド: GET

残り回数: 4

有効期限: Apr:24:00:00:00:1998:GMT, Apr:25:00:00:00:1998:GMT

ケイパビリティ識別子: Rnd2

メソッド: GET

残り回数: 1

有効期限: Apr:24:10:00:48:1998:GMT, Apr:25:10:05:48:1998:GMT

【0201】そして、プロキシ・オブジェクト301は、該当エントリの履歴の内容を以下のように変更して

から、対応するHTTPオブジェクト391に対してメッセージGETを送信する。ここで行われるケイパビリ

ティの変更内容は、残り回数を1だけ減分することである。

【0202】

【数23】

ケイパビリティ識別子: Rnd1

メソッド: GET

残り回数: 3

有効期限: Apr:24:00:00:00:1998:GMT, Apr:25:00:00:00:1998:GMT

ケイパビリティ識別子: Rnd2

メソッド: GET

残り回数: 0

有効期限: Apr:24:10:00:48:1998:GMT, Apr:25:10:05:48:1998:GMT

【0203】この後、プロキシ・オブジェクト301は、世界時で10時5分48秒になった時点で、ケイパビリティ・リスト保持部331中の識別子Rnd2に対応するエントリを削除する（削除しないまでも、エントリに対応するケイパビリティは失効する）。

【0204】次に、HTTPクライアント50Cが他のHTTPクライアント50Dに与えたケイパビリティcap2を無効化する処理について説明する。

※【0205】この場合、HTTPクライアント50C

は、アクセス識別子aと、【数11】に示したケイパビリティcap1と、HTTPクライアント50Dに与えたケイパビリティに含まれるケイパビリティ識別子Rnd2を基に、下式【数24】のようなメッセージを生成して、HTTPサーバ300に送信する。

【0206】

【数24】

GET /a/revocation/cap1,cap2 HTTP/1.1

【0207】このメッセージは、aというアクセス識別子を持つアクセス権限保持部321に対して、ケイパビリティcap1から派生したケイパビリティcap2の無効化を要求する旨の内容を含んでいる。

【0208】これに対し、プロキシ・オブジェクト30★

★1は、以下のような《処理手順3》を実行して、ケイパビリティcap2がケイパビリティcap1から派生したものであるかを検証することができる。

【0209】

【数25】

《処理手順3》

1: ケイパビリティcap1の変更履歴をhistory1とする。

2: ケイパビリティcap2の変更履歴をhistory2とする。

3: history1がhistory2の先頭からの連続する部分として含まれるかどうかを調べる。含まれれば次ステップに進み、そうでなければ当該ケイパビリティは派生でないとして、本処理手順を終了する。

4: ケイパビリティcap2の正当性を《処理手順1》を用いて検証する。正しければ当該ケイパビリティは派生であるとして無効リストにケイパビリティcap2を追加し、正しくなければ派生ではないとして本処理手順を終了する。

【0210】プロキシ・オブジェクト301は、無効化要求に含まれる第2のケイパビリティcap2が第1のケイパビリティcap1の派生である場合には、要求に含まれるアクセス識別子の無効リスト保持部341に当

該ケイパビリティcap2を追加登録する。

【0211】なお、無効化されたケイパビリティのうち、そもそも権限の有効期限により無効となるものについては、無効リストから取り除いてもよい。



【0212】〔追補〕以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

#### 【0213】

【発明の効果】以上詳記したように、本発明によれば、ネットワーク上に複数のコンピュータ・システムが相互接続された分散型のネットワーク・システムにおいて、ネットワーク上に存在するオブジェクトを安全に保護することができる、優れたオブジェクトのアクセス管理方式を提供することができる。

【0214】また、本発明によれば、オブジェクトのアクセス権限を記述したケイパビリティ (Capability) をクライアントに配ることによってオブジェクトへのアクセスを許可するタイプの、優れたアクセス管理方式を提供することができる。

【0215】また、本発明によれば、ケイパビリティを保有するクライアントが権限を変更したケイパビリティを自由に生成し、他のクライアントに安全に委譲することができる、優れたオブジェクトのアクセス管理方式を提供することができる。

【0216】また、本発明によれば、ケイパビリティを保有するクライアントが権限を変更したケイパビリティ\*

\*を自由に生成し、オブジェクトを管理するサーバは生成されたケイパビリティを安全に検査することができる、優れたオブジェクトのアクセス管理方式を提供することができる。

【0217】また、本発明によれば、ケイパビリティを保有するクライアントが権限を変更したケイパビリティを自由に生成し、さらに生成したケイパビリティを確実に無効化することができる、優れたオブジェクトのアクセス管理方式を提供することができる。

#### 【図面の簡単な説明】

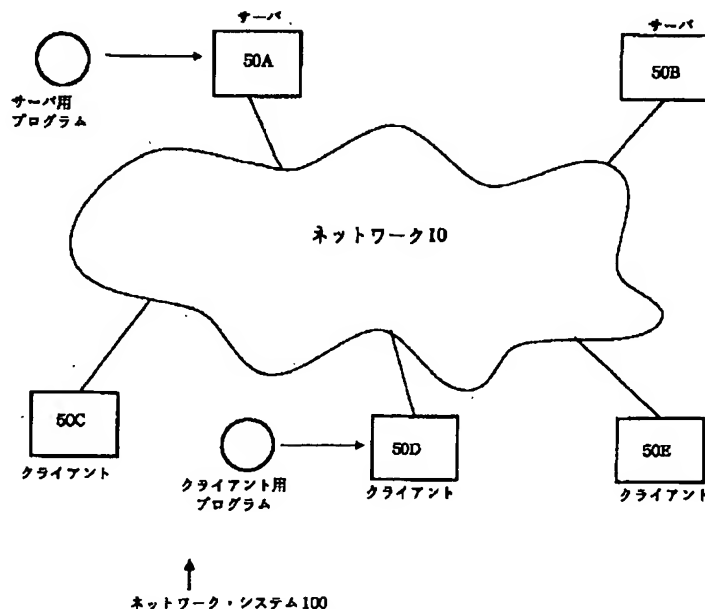
【図1】 本発明の実施に供されるネットワーク・システム100の構成を模式的に示した図である。

【図2】 オブジェクト・サーバ50Aの構成を模式的に示した図である。

#### 【符号の説明】

10…ネットワーク  
50…コンピュータ・システム  
100…ネットワーク・システム  
300…HTTPサーバ  
301…プロキシ・オブジェクト  
302…アクセス権限集合  
311, 31M…アクセス権限保持部  
321, 32M…アクセス識別子保持部  
331, 33M…ケイパビリティ・リスト保持部  
341, 34M…無効リスト保持部  
351, 35M…リファレンス保持部  
391, 39M…HTTPオブジェクト

【図1】



【図2】

